[FSW Test Procedure Template]

[A test procedure is a STOL (or whatever scripting language is used on the XXX Mission) script that can be executed in the FSW test facility and that implements a test scenario.

When developing test procedures, the developer shall adhere to the following guidelines:

- Test procedures shall be automated and self-checking to the maximum extent possible. Requirement pass/fail status shall be automated within test procedures. The goal is for fully unattended test execution and automated reporting of results without compromising accuracy of the verification or severely impacting productivity (i.e., if it's going to take three weeks of programming effort to automate something that can be done in ten minutes manually, do it manually).
- To the extent possible the test procedure developer shall use the FSW user documentation when developing the test procedures to verify the accuracy and completeness of the documentation for flight operations.
- The test procedure developer shall carefully study the scenario upon which the procedure is to be based. The goal of each scenario is to test some function of the flight software. The developer shall pay special attention to how the scenario attempts to exercise the various aspects of that function and be careful to verify that each aspect was exercised correctly. If necessary, the developer shall ask the scenario author for any needed clarifications.
- In addition to writing the test procedure, the developer shall also develop any other supporting procedures, simulator configuration files [sometimes called namelists], and input data files (such as system tables) that are necessary for implementing the scenario.
- Test procedures, subprocedures, and supporting files shall adhere to the following naming convention: [Supply a naming convention.]
- The commands and telemetry needed to support the test are identified from the Ground Telemetry and Command Database and the test procedure is drafted in accordance with scripting language reference manual. The test procedure developer shall use the ground system and FSW test facility user documentation to create the proper conditions for the test.
- Tests will use normal ground operation capabilities, documentation, and procedures routinely used by operations personnel as much as possible.
    - The FSW test facility shall be used for real-time commanding and memory loads. It shall also be used to monitor real-time telemetry, the primary source of test results for analysis.
    - Test developers shall avoid using special testing tools or non-operational techniques for testing the FSW. This may include such things as bus monitors, debuggers, or using FCP to set or dump FSW variables.
    - Test developers shall refer to commands by command mnemonic rather than using the hexadecimal version of the command. The only exception should be when the Test developer deliberately attempts to send a non-defined command in order to test an error condition.
- Operational stored command sequences will be used if available, otherwise place holder stored command sequences with enough functionality to perform the test may be developed and used.

All test procedures shall be repeatable with clear initial conditions. All test procedures and their output will be archived in order to facilitate troubleshooting and regression test comparisons. In addition, all tables, which were loaded as part of the procedure, will be archived.

This standard specifies the required minimum set of procedure elements. Although each element must be present in the procedure, the syntax of the elements is dependent on the scripting language. The test scenario document should be copied as the starting document for the test procedure. Comment lines copied from the test scenario should be updated as needed and the test language inserted.

The Test Lead must identify a test procedure naming standard in order to ensure that test procedure names are descriptive, consistent, recognizable, and uniform for each FSW build. See the Test Description Document Template for the test procedure naming standard.

As you fill out this template, remove this page and the blue tailoring advice text.

Template Update History

Version	Date	Description	Affected Pages
1.0	02/23/04	Original	All
1.1	01/09/06	Final changes based on 01/03/06 review by Test Team Leads.	All

The template begins on the next page.]

```
[ PROC procedurename ]
;*****************************************************************
;  Test Name:  [ Provide test procedure name ]
;  Test Level: [ Identify test levels.  Test levels include:  Build
;               Verification, System Validation ]
;  Test Type:  [ Identify test type.  Test types include:  External
;               Interface, Functional, Performance, Load/Stress, Long
;               Duration, Regression ]
;  Test Description
;     [ Enter a short paragraph that states the test purpose, test
;      objectives, and provides a brief description of the test(s).
;      Identify each test case. ]
;
;  Requirements Tested
;     [Provide Software Requirement numbers and (optionally) text. ]
;
;  Prerequisite Conditions
;     [ Identify any prerequisite conditions.  See guidance in the Test
;      Scenario Template. ]
;
;  Assumptions and Constraints
;     [ Identify any assumptions made and constraints or limitations.
;      See guidance in the Test Scenario Template. ]
;
;  Change History
;    [ Provide an outline of the development history for the procedure.
;     Under Date, specify the month, day, and year of the change. Under
;     Name, specify the first initial and the last name of the author
;     for the current implementation or change to the procedure. Under
;     Description, briefly describe the change. For the first entry of
;     a new procedure, specify “original procedure” under Description.
;     Change history should start following run for record. ]
;
;     Date           Name           Description
;     [ 08/01/03 ]   [ A. Smith ]   [ Original Procedure ]
;
;  Arguments
;     [ Specify the name and description of the calling procedure
;      variables.  Under Name, specify the variable name as specified
;      in the calling sequence.  Under Description, specify the purpose
;      of the variable and relevant information (e.g. flag values). ]
;
;     Name              Description
;     [ Name ]          [ Purpose of variable and relevant information.
;                         Include units (e.g., feet, degrees]
```

```
;
;  Procedures Called
;     [ Specify the name and description of the procedures that will be
;       called by the test.  Under Name, specify the procedure name.
;       Under Description, specify the purpose of procedure. ]
;
;     Name              Description
;     [ Proc called ]   [ Purpose of procedure]
;
;  Required Post-Test Analysis
;     [ Identify post-test data reduction and
;       analysis, as applicable.  See Test Scenario Template
;       guidelines. ]
;*******************************************************************
;*******************************************************************
;  Open archive and event log file
;*******************************************************************
[ LOG ENABLE ]
[ ARCHIVE OPEN ]

;*******************************************************************
;  Define variables
;*******************************************************************
[ GLOBAL ]
[ LOCAL ]

;*******************************************************************
;  Get start time and write time to the log
;*******************************************************************

[ LOCAL S_TIME  ]

[ EVENT CLEAR ]
[ S_TIME = GMT() ]
[ WRITE "STARTING PROCEDURE - <procedure_name>", $S_TIME ]




;*******************************************************************
;  Record software version numbers
;*******************************************************************

;*******************************************************************
;  Run initialization procedure [ if necessary ]
;*******************************************************************
```

```
WRITE "*****************************************************************"
WRITE " Step 1:  [ Enter description of test step.  This description  "
WRITE "            may be copied from the test scenario.  As          "
WRITE "            applicable, include requirements verified, expected"
WRITE "            results and evaluation criteria.  Add additional   "
WRITE "            steps as needed. ]                                 "
WRITE "*****************************************************************"

WRITE "*****************************************************************"
WRITE " Step 1.1:  [ Enter description of test substep.  This         "
WRITE "              description may be copied from the test scenario."
WRITE "              As applicable, include requirements verified,    "
WRITE "              expected results and evaluation criteria.  Add   "
WRITE "              additional substeps as needed. ]                 "
WRITE "*****************************************************************"

;*****************************************************************
;  Get stop time and write time to the log
;*****************************************************************
[ S_TIME = GMT () ]
[ WRITE "END OF PROCEDURE – Enter procedure name here", GMT()]

;*****************************************************************
;  Calculate and write procedure duration to the log
;*****************************************************************
[ WRITE "DURATION OF – Enter procedure name here     ", GMT()-$S_TIME ]

;*****************************************************************
;  Save the event logs and close archive
;*****************************************************************
[ ARCHIVE CLOSE ]
[ LOG DISABLE ]
;*****************************************************************
;  End procedure [ Enter procedure name ]
;*****************************************************************
[ ENDPROC ]
```